Attorney Docket No: IDF 1398 (4000-00700)   *Patent*

REMARKS

Status of Claims

Claims 1-20 are currently pending, comprising original claims 1-18 and new claims 19-20.

Response to Prior Art Rejection

Claims 1-6 stand rejected under 35 USC 103(a) as obvious over *Klein* (US 5,835,763), and claims 7-18 stand rejected under 35 USC 103(a) as obvious over *Klein* in view of *Swartz* (US 6,625,651). Applicants respectfully submit that the art of record does not establish a prima facie case of obviousness as to the pending claims. According to MPEP 2142, three basic criteria must be met to establish a *prima facie* case of obviousness:

> First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure.

The Examiner has failed to establish a prima facie case of obviousness as *Klein*, alone or in combination with *Swartz* (assuming for the sake of argument that such is proper), does not teach or suggest all of the claim limitations.

Klein, the primary reference, discloses a method for a computer operating in a non-threaded (i.e., synchronous) environment to process batch jobs in a threaded (i.e., asynchronous) manner, which is achieved via a complicated threading and queuing array as shown in Fig. 1 of *Klein*. Thus, *Klein* discloses the asynchronous processing of batch jobs (see e.g., col. 3, lines 12-15), meaning that an application can effectively have multiple batch jobs running at the same time and thereby efficiently use computing resources.

In contrast, Applicants' claimed invention does not relate to the conversion of synchronous processing of batch jobs to asynchronous processing of batch jobs. Instead, Applicants' claimed invention relates to efficient reuse of programming code and platform independence by encapsulating a given batch job and providing a uniform application

5

17286.02/4000.00700

programming interface (API) for applications processing the batch job (see e.g., page 1, line 19 to page 2, line 2).

In order to further demonstrate the differences between Applicants' claimed invention and *Klein*, Applicants provide herewith Exhibit A showing a hypothetical combination wherein Applicants' claimed invention as illustrated by Fig. 2A and page 3, lines 7-8 of the specification is used with the asynchronous conversion method disclosed by Fig. 1 of *Klein*. The thread enabling layer 12 of *Klein* may process a batch job directly, as shown by thread 26 calling batch job 18. Alternatively, the thread enabling layer 12 may process a batch job via the encapsulation functionality provided by Applicants' claimed invention, as shown by thread 25 invoking batch framework 10 to process batch job 15. In an embodiment where the *Klein*'s synchronous to asynchronous conversion functionality is not needed, an application 20 can directly invoke batch framework 10 to process batch job 15. As is clearly shown by Exhibit A, *Klein* discloses complimentary functionality which may be used in combination with, rather than in place of, Applicants' claimed invention.

In view of the fundamental differences between *Klein* and Applicants' recited invention as explained above, it should be readily apparent that *Klein*, alone or in combination with *Swartz*, does not teach or suggest all of the recited claim limitations. Applicants note with appreciation the Examiner's acknowledgement that *Klein does not teach using classes to dispatch batch jobs*. The use of a class comprising a method to execute the batch job is fundamental to the architecture and functionality of Applicants' claimed invention, and the absence of such an important element cannot be readily dismissed. *Klein* does not teach or suggest the use of a class comprising a method to execute the batch job – in contrast, *Klein* employs a complicated array of threads and queues to achieve an entirely different purpose, namely the conversion of synchronous processing to asynchronous processing. In fact, *Klein* does not even contain the term "class" anywhere in the specification. The Examiner points out that *Klein* teaches a method to execute a batch job, but such a method is employed in the overall thread/queue architecture and in now way relates to using classes to dispatch batch jobs. The Examiner further asserts that *Klein* teaches using object in object-oriented programming, but Applicants respectfully traverse that *Klein* can reasonably be interpreted as teaching such, and certainly not in the context of

6

Applicants' claimed invention. Specifically, the Examiner relies on col. 10, lines 10-11, which reads:

> PgmObj—This input parameter defines the name of a CL
> command, or a program object, that the thread will act on.

Facially, this isolated passage is woefully inadequate in establishing that *Klein* teaches or suggests each and every element of Applicants' claimed invention. This passage teaches nothing regarding the use of a class comprising a method to execute a batch job, and the Examiner provides no explanation as to how this passage teaches or is equivalent to any of the elements recited in Applicants' claims. Furthermore, this passage does not provide a requite suggestion or motivation to alter the fundamental thread/queue architecture of *Klein* to arrive at Applicants' claimed invention. Likewise, the Examiner's reliance on the fact that classes are a standard in object oriented programming does not teach or suggest the specific recited combination in Applicants' claims, nor does it provide the requisite suggestion or motivation to alter the fundamental thread/queue architecture of *Klein* to arrive at Applicants' claimed invention. Finally, *Klein* does not teach or suggest a scheduler or its recited functionality of invoking the batch framework according to a predetermined schedule.

In summary, Applicants respectfully submit that fundamental differences exist between *Klein* and Applicants' claimed invention and that important elements recited in Applicants' claims are completely absent from and not suggested by the disclosure of *Klein*. Therefore, Applicants respectfully submit that *Klein* does not establish a prima facie case of obviousness as to independent claim 1. Given that claims 2-20 dependent from claim 1, the deficiencies of the primary reference, *Klein*, as described previously, likewise preclude a prima facie case of obviousness as to claims 2-20 in view of *Klein*, alone or in combination with *Swartz*.
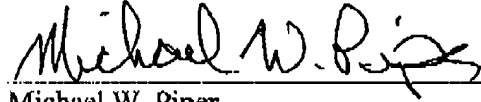
7

CONCLUSION

The Commissioner is hereby authorized to charge payment of any further fees associated with any of the foregoing papers submitted herewith, or to credit any overpayment thereof, to Deposit Account No. 21-0765, Sprint.

Applicants respectfully submit that the present application as amended is in condition for allowance. If the Examiner has any questions or comments or otherwise feels it would be helpful in expediting the application, he is encouraged to telephone the undersigned at (972) 731-2288.

Respectfully submitted,

Date: 6/11/2004

CONLEY ROSE, P.C.
5700 Granite Parkway, Suite 330
Plano, Texas 75024
(972) 731-2288
(972) 731-2289 (facsimile)

Michael W. Piper
Reg. No. 39,800

ATTORNEY FOR APPLICANTS

8

17286.02/4000.00700

App. No. 09/698,729
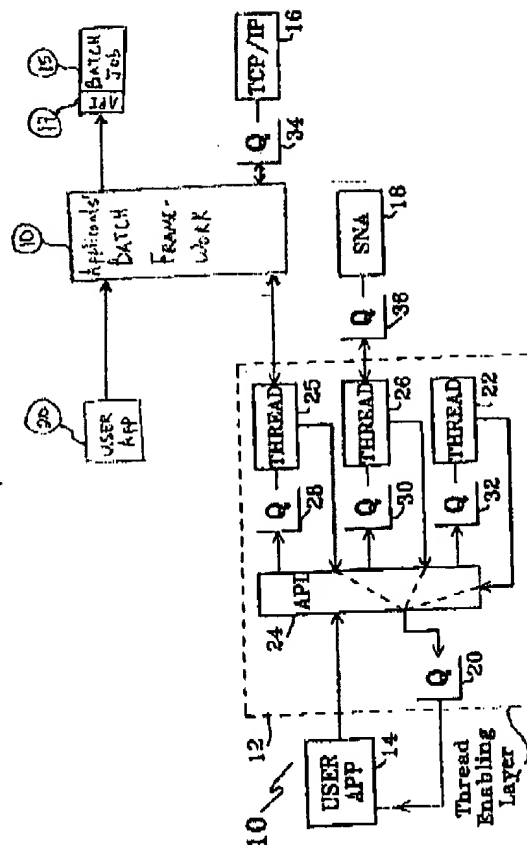
Applicants' Fig. 2A; page 3, lines 7-8

Figure 1

of Klein

Exhibit A - Hypothetical combination of Applicants' claimed invention with prior art to show differences therebetween